طراحی کامپیوتری سیستم‌های دیجیتال

جلسه دوم



Decimal	Binary	Octal	Hexadecimal
00	00000	00	00
01	00001	01	01
02	00010	02	02
03	00011	03	03
04	00100	04	04
05	00101	05	05
06	00110	06	06
07	00111	07	07
08	01000	10	08
09	01001	11	09
10	01010	12	0A
11	01011	13	0B
12	01100	14	0C
13	01101	15	0D
14	01110	16	0E
15	01111	17	0F
16	10000	20	10
17	10001	21	11
18	10010	22	12

مبنا

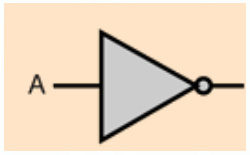
مبنای دودویی Binary

مبنای هشت Octal

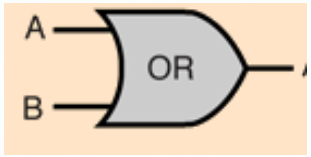
مبنای ده دهی Decimal

مبنای شانزده Hexadecimal

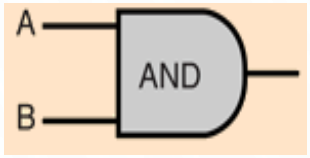
جدول درستی Truth Table



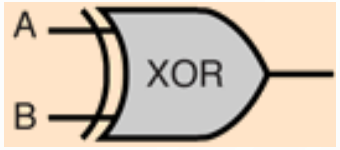
IN	OUT
0	1
1	0



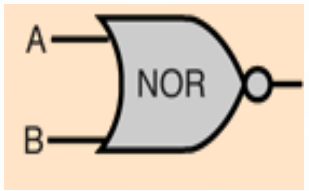
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1



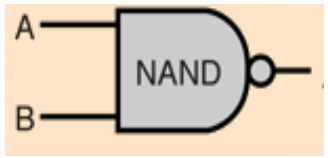
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1



A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0



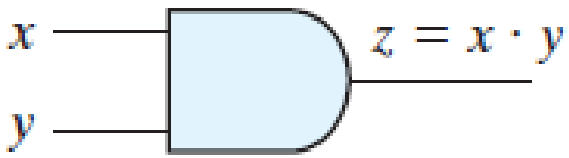
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0



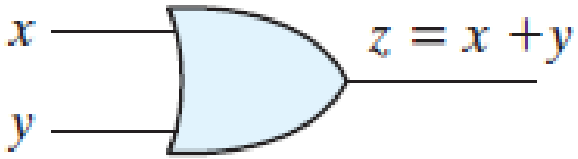
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

منطق دودویی در گیت های منطقی

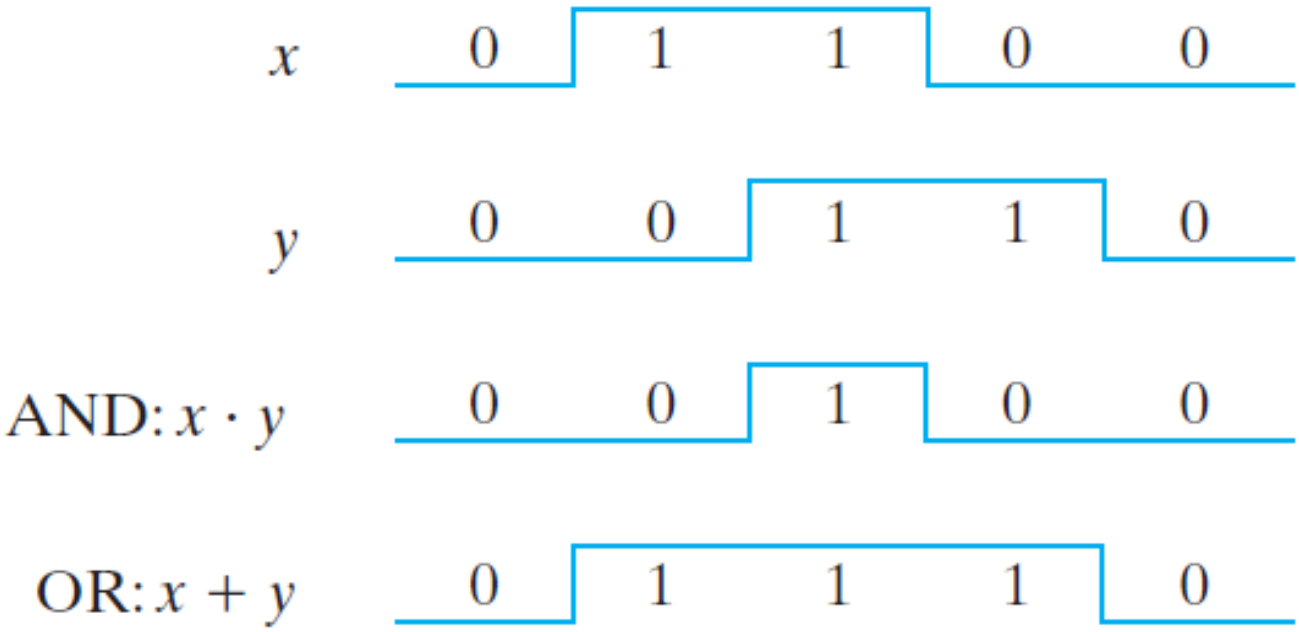
ورودی و خروجی بر اساس سیگنال صفر و یک



Two-input AND gate

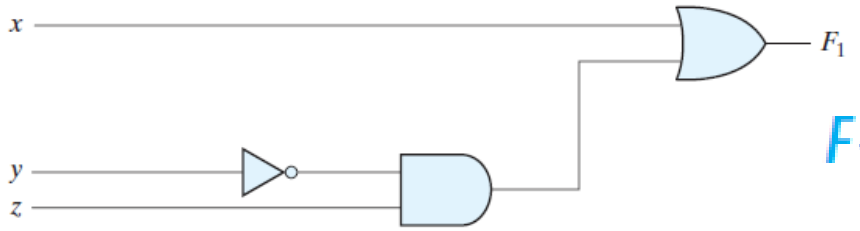


Two-input OR gate

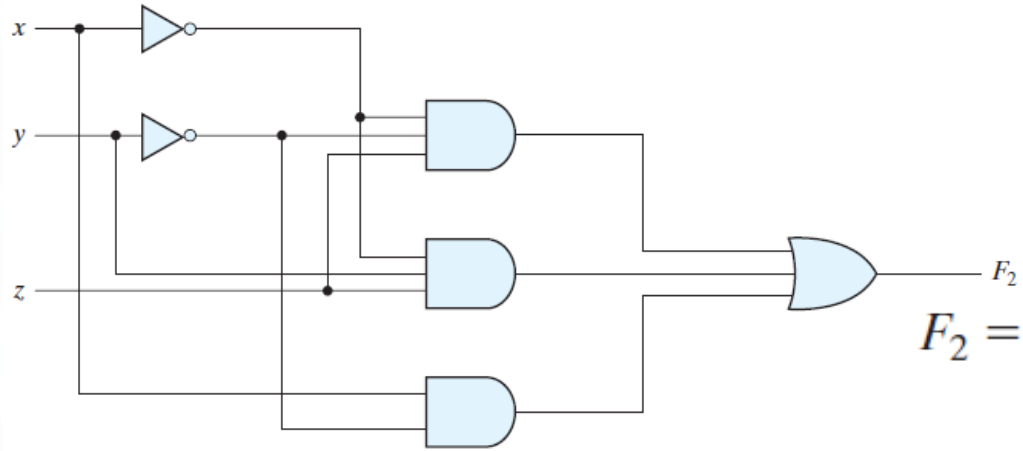


مدار منطقی

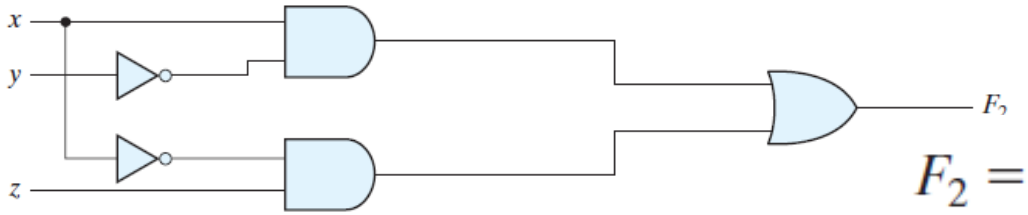
خروجی یک مدار منطقی



$$F_1 = x + y'z$$



$$F_2 = x'y'z + x'yz + xy'$$



$$F_2 = xy' + x'z$$

زبان Verilog HDL

طراحی سیستم های دیجیتال در این زبان به دو فرم کلی زیر انجام می شود:

- توصیف ساختاری Structural Description
- توصیف رفتاری Behavioral Description

طراحی و تحقیق درستی آن نیازمند انجام موارد زیر است:

- توصیف سیستم (مدل کردن طرح)
- کامپایل compile وبررسی ساختار دستوری مدل Syntax
- شبیه سازی مدل توصیف شده و تحقیق درستی آن

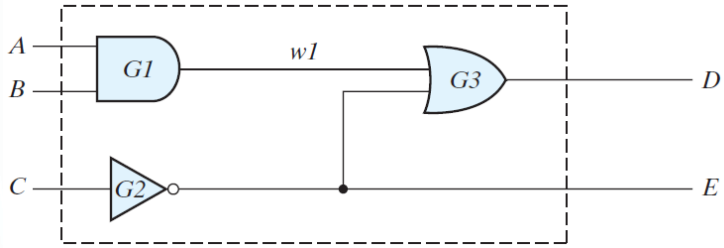
استانداردها

اولین استاندارد در سال 1995 و نسخه های ارتقا یافته در سالهای 2001 و 2005 تصویب شد.

منطق ترکیبی

پیاده سازی منطق ترکیبی در Verilog

- توصیف یک مجموعه اتصالات شماتیکی از گیت ها



- توصیف یک مجموعه از معادلات بولی

$$E = A + BC + B'D$$
$$F = B'C + BC'D'$$

- توصیف از طریق جدول حقیقت

x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

زبان Verilog HDL

کلمات کلیدی این زبان که حدود 100 تا است با حروف کوچک از پیش تعریف شده است.

کلمات کلیدی ساخت های زبان را توصیف میکند.

علامت // برای توضیحات یک خطی استفاده می شود.(سبز رنگ)

علامت */ برای شروع و علامت /* برای پایان توضیحات چند خطی استفاده می شود.

کلمات کلیدی در زبان Verilog

فهرست کلمات کلیدی

always	endmodule	large	reg	
and	endprimitive	macromodule	release	tranif0
assign	endspecify	nand	repeat	tranif1
attribute	endtable	negedge	rnmos	tri
begin	endtask	nmos	rpmos	tri0
buf	event	nor	rtran	tri1
bufif0	for	not	rtranif0	triand
bufif1	force	notif0	rtranif1	trior
case	forever	notif1	scalared	trireg
casex	fork	or	signed	unsigned
casez	function	output	small	vectored
cmos	highz0	parameter	specify	wait
deassign	highz1	pmos	specparam	wand
default	if	posedge	strength	weak0
defparam	ifnone	primitive	strong0	weak1
disable	initial	pull0	strong1	while
edge	inout	pull1	supply0	wire
else	input	pulldown	supply1	wor
end	integer	pullup	table	xnor
endattribute	join	rcmos	task	xor
endcase	medium	real	time	
endfunction	module	realtime	tran	

Verilog -Module

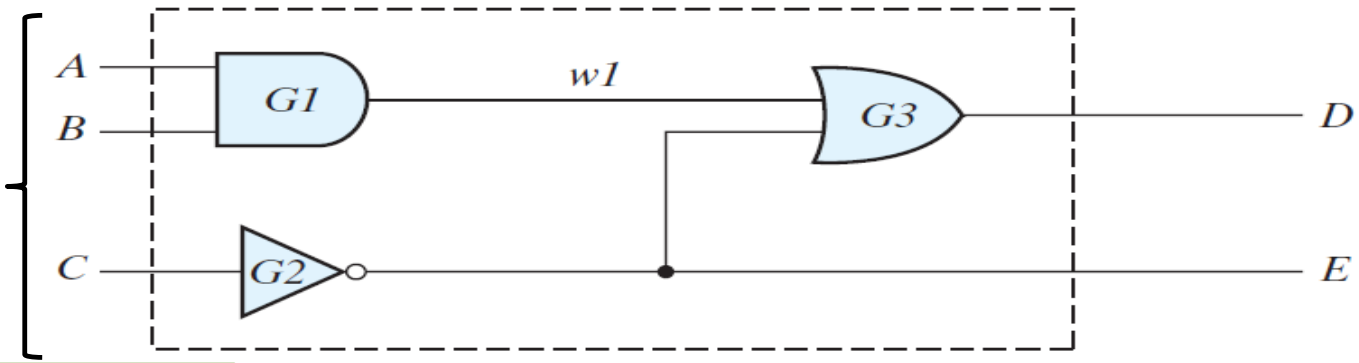
اولین و اصلی ترین واحد توصیفی این زبان "ماژول" است.

هر ماژول دارای یک نام و تعدادی ورودی و خروجی است.

پایان هر ماژول نوشتن endmodule الزامی است.




```
module Simple_Circuit (A, B, C, D, E);
```




```
endmodule
```

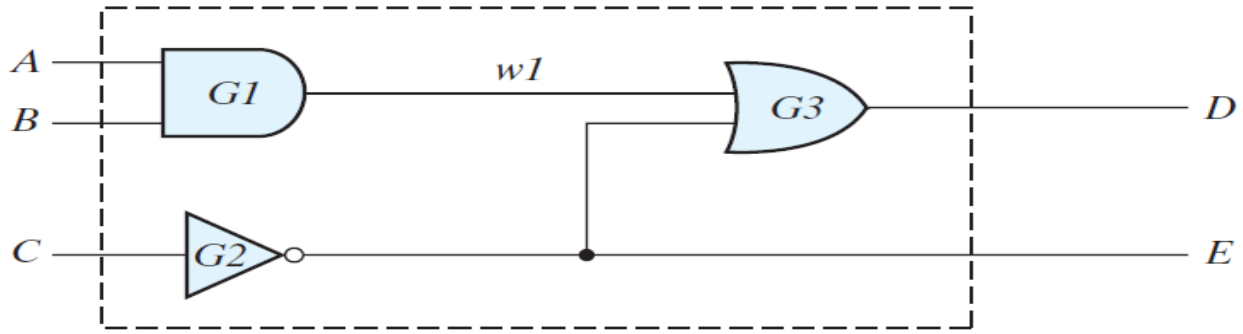
اولین برنامه

توصیف اتصالات شماتیکی ، توصیف در سطح گیت

```
// Verilog model of circuit of Figure 3.35. IEEE 1364–1995 Syntax

module  Simple_Circuit (A, B, C, D, E);
  output        D, E;
  input         A, B, C;
  wire          w1;

  and           G1 (w1, A, B); // Optional gate instance name
  not           G2 (E, C);
  or            G3 (D, w1, E);
endmodule
```



Verilog- Wire

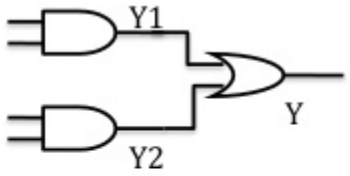
Wire ها نه ورودی هستند و نه خروجی و نمی توان آنها را به عنوان پورت های ورودی یا خروجی تعریف کرد. پس این نوع اتصالات را به عنوان سیم یا اتصال داخلی تعریف می کنیم.

فقط نام پورت ها/ درگاه ها (ورودی و خروجی ها) در اعلان یک ماژول لیست می شود.

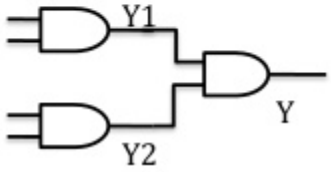
اتصالات داخلی می تواند یکی از انواع زیر باشد:

(wire, wand, wor, tri, triand, trior, trireg, etc)

```
wor Y ;        // or gate for Y1 & Y2
```




```
wand Y;        // and of Y1& Y2
```



نکته: داخل **"میز تست"** خروجی ها با کلمه کلیدی wire اعلام می شود.

Verilog-syntax

بررسی نحوی برنامه

```
module Simple_Circuit (A, B, C, D, E);
  output   D, E;
  input    A, B, C;
  wire     w1;

  and      G1 (w1, A, B); // Optional
  not      G2 (E, C);
  or       G3 (D, w1, E);
endmodule
```

نامگذاری گیت های بنیادی اختیاری است

شناسه در وری لاگ

شناسه ها : نام هایی که به ماژول ها ، متغییرها و سایر عناصر زبان داده می شود.

وجود شناسه (نام) برای گیت های بنیادی اختیاری است.

not G2(E,C) = not (E,C)

شناسه ها از نویسه های الفبا عددی همراه با (-) تشکیل شده است.

شناسه ها باید با یک نویسه الفبایی همراه با (-) شروع شود.

شناسه ها نمی توانند با عدد شروع شوند.

```
module Circuit_with_UDP_02467 (e, f, a, b, c, d);
```

خروجی یک جز بنیادین باید ابتدا فهرست شود ولی ورودی و خروجی های یک ماژول می تواند به هر ترتیبی فهرست شود.

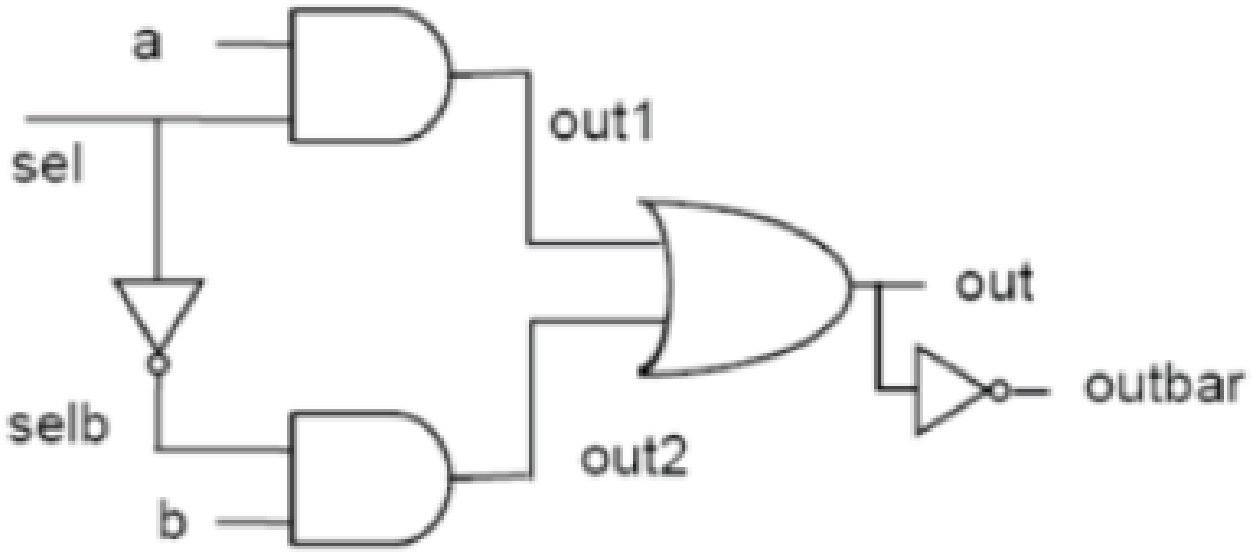
وری لاگ به بزرگی و کوچکی حروف حساس است (AND همان and نیست).

گیت های بنیادی با کلمات کلیدی برای این زبان تعریف شده و نیازی به توصیف عملکرد آنها نیست.

and, or, nor, xor, nand, xnor, buf, not

توصیف در وری لاگ

مدار زیر را به زبان وریلاگ توصیف کنید.



جواب

```
module Exercise1_(a,sel,b,out,outbar);
input a,b,sel;
output out,outbar;
wire selb,out1,out2;

and (out1,a,sel);
and (out2,b,sel);
or (out,out1,out2);
not (outbar,out);
not (selb,sel);

endmodule
```

تاخیر انتشار

زمان مورد نیاز برای گذر یک ورودی و رسیدن آن به خروجی

تاخیر های زمانی در Verilog بدون بعد هستند.

استفاده از مقیاس زمانی:

```
`timescale 1ns/100ps
```

واحد اندازه گیری برای تاخیر های زمانی (1ns)

دقتی که واحد زمانی بر اساس آن گرد می شود (100 ps)

تاخیر انتشار

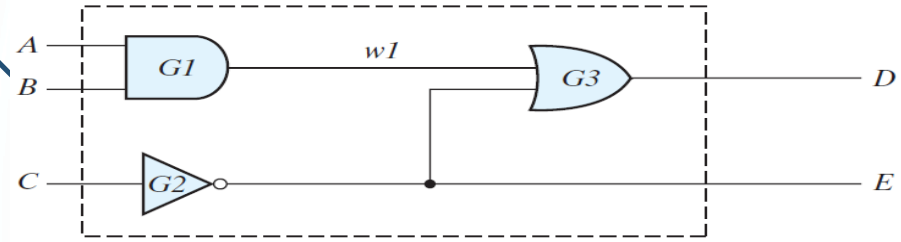
مدل سطح گیت همراه با تاخیر انتشار

```
// Verilog model of simple circuit with propagation delay

module Simple_Circuit_prop_delay (A, B, C, D, E);
  output D, E;
  input  A, B, C;
  wire   w1;

  and          #(30) G1 (w1, A, B);
  not          #(10) G2 (E, C);
  or           #(20) G3 (D, w1, E);
endmodule
```

تاخیر انتشار




```
and     #(30) G1 (w1, A, B);
not     #(10) G2 (E, C);
or      #(20) G3 (D, w1, E);
```

ورودی A=0, B=0 ,C=0

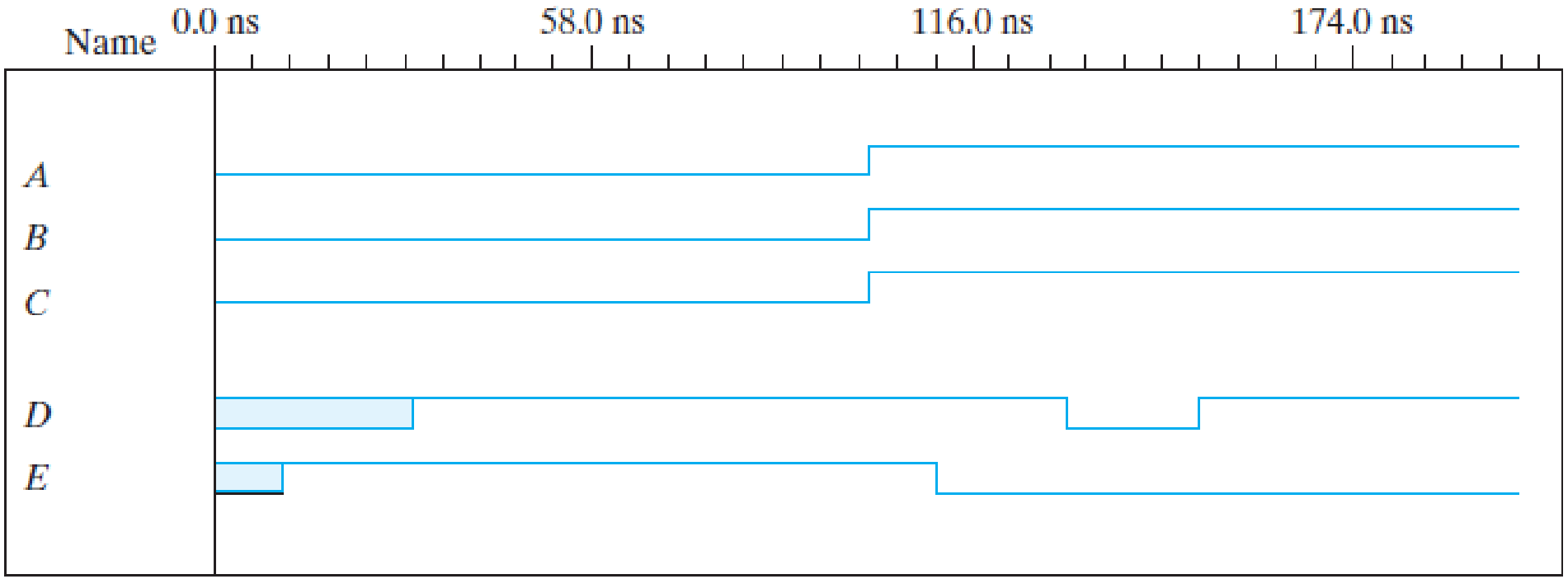
تغییر A=1, B=1, C=1

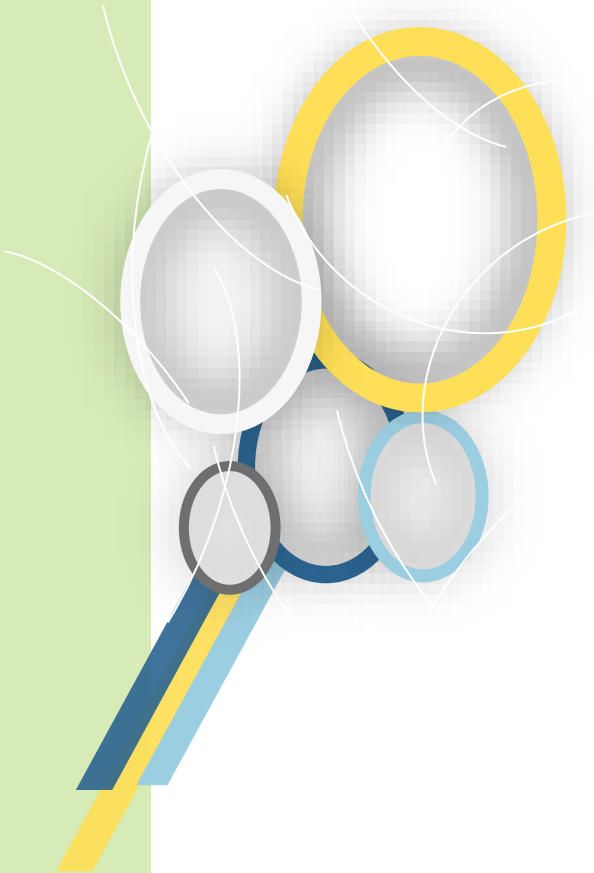
Time Units (ns)	Input ABC	Output E w1 D
—	0 0 0	1 0 1
—	1 1 1	1 0 1
10	1 1 1	0 0 1
20	1 1 1	0 0 1
30	1 1 1	0 1 0
40	1 1 1	0 1 0
50	1 1 1	0 1 1

تاخیر انتشار

ورودی A=0, B=0 ,C=0

پس از 100ns تغییربه A=1, B=1, C=1





خسته نباشید

تمرین -1

خطا های نحوی syntax errors برنامه زیر را پیدا کنید.

```
module Exmpl-3(A, B, C, D, F)                    // Line 1
  inputs        A, B, C, Output D, F,            // Line 2
  output        B                                // Line 3
  and           g1(A, B, D);                     // Line 4
  not           (D, A, C),                       // Line 5
  OR            (F, B; C);                       // Line 6
endmodule;                                       // Line 7
```

تمرین-2

مدار منطقی مربوط به کد ریز را رسم کنید

```
module Circuit_A (A, B, C, D, F);
  input      A, B, C, D;
  output     F;
  wire       w, x, y, z, a, d;
  or         (x, B, C, d);
  and        (y, a ,C);
  and        (w, z ,B);
  and        (z, y, A);
  or         (F, x, w);
  not        (a, A);
  not        (d, D);
endmodule
```